TODO: My Thesis Title

Jesse Wu

A subthesis submitted in partial fulfillment of the degree of
Bachelor of Computer Science (Honours) at
The Research School of Computer Science
Australian National University

July 2014

© Jesse Wu

Typeset in Palatino by TEX and LATEX 2_ε.

Except where otherwise indicated, this thesis is my own original work.

Jesse Wu
11 July 2014

To fluffy the cat.

Acknowledgements

Here is where you thank the people who helped you along the way…

Abstract

This should be the abstract to your thesis…

Contents

Acknowledgements **vii**

Abstract **ix**

1 An Introduction to My Thesis **1**
1.1 The Basis for this Work 1
1.1.1 A Theoretical Framework 1

2 Logic **3**
2.1 Introduction 3
2.2 Validity 3
2.3 Sequent Calculi 3

3 Cut Elimination **5**
3.1 Motivation 5
3.2 Object vs Meta Level Reasoning 5
3.3 Cut Elimination Proofs 5

4 Interactive Theorem Proving **7**
4.1 Current Provers 7
4.2 Shallow and Deep Embeddings 7

5 Formalisation for KM.LC **9**
5.1 Representing Derivations 9
5.2 TODO:STUFF 9

6 Automated Theorem Proving **11**
6.1 IntHistGC 11
6.1.1 Global Caching 11
6.2 IntLSJ 11

7 Conclusion **13**
7.1 Why this is a Very Clever Thesis 13

A Appendix **15**
A.1 Code 15

Chapter 1

An Introduction to My Thesis

In this thesis:

A brief introduction to the theoretical side of proof systems for logic, with a focus on Intuitionistic Logics.

A description of a formalisation of cut elimination for KM.LC in Isabelle.

We also dedicate a section to automated theorem proving.

1.1 The Basis for this Work

1.1.1 A Theoretical Framework

Chapter 2

Logic

2.1 Introduction

Basic introduction to logic - variables, boolean values, connectives etc.

Include uses - applications etc.

A formula is built up from a number of atomic propositions and a number of logical connectives.

We assume the reader is familiar with the syntax and semantics for classical logic.?

2.2 Validity

How to determine whether a formula is valid within a particular logic.

Introduce various types of calculi. Tableaux, Sequent, Natural Deduction.

How do these work?

A sequent is an expression $\Gamma \vdash \Delta$ where both Δ and Γ are multisets of formulae.

2.3 Sequent Calculi

Give basic calculus (probably LK) and explain how rules/backwards proof search works. Example derivations.

Chapter 3

Cut Elimination

TODO: have a chapter/section just about rule admissibility before cut elimination? Could have pen and paper proofs for things like the admissibiliy of weakening as an example.

Cut is special structural rule.

3.1 Motivation

Why does cut exist in the first place?

Why remove cut? Subformula property, backwards proof search, interpolation etc

3.2 Object vs Meta Level Reasoning

Sequent calculi are used to prove things at the object level - validity of formulae.

At the meta level we reason about the calculi themselves, and the proofs that they produce.

3.3 Cut Elimination Proofs

Highly technical

Publications being released proving cut for various calculi. These are generally pen and paper proofs. Due to the high number of cases involved these proofs can easily have mistakes.

People are notoriously bad at doing repetitive tasks, with minor alterations. ”The other case is similar” is commonly used for cases in a cut elimination proof.

Give an example.

Chapter 4

Interactive Theorem Proving

Interactive theorem provers allow for the creation of computer checked proofs. While they do not entirely remove the human element - a proof developed with such software will not be susceptible to errors common in pen and paper proofs.

Introduce how these provers work - based on logic, core code heavily scrutinised.

4.1 Current Provers

What software is available. Advantages and disadvantages of each.

Talk about using abella at first.

Why pick Isabelle.

Learning to use an ITP is similar to learning a new programming language.

4.2 Shallow and Deep Embeddings

In order to perform reasoning about a calculus using an ITP, one must first encode that calculus within the language of the ITP. There are two different ways of doing so, which are generally referred to in the literature as either ”shallow” or ”deep” embeddings. The difference between the two is somewhat analogous to the difference between object and meta level reasoning.

Shallow embeddings: represent formula directly in logic of ITP.

Deep embeddings: encode syntax for formula, etc. Allows representation of derivation trees.

Chapter 5

Formalisation for KM.LC

5.1 Representing Derivations

5.2 TODO:STUFF

Chapter 6

Automated Theorem Proving

While our computer assisted proof of cut elimination for KM.LC forms the main focus of this thesis, we also developed some automated theorem provers for Intuitiontistic logic. These essentially constitute a foray into object level reasoning and implementation, as opposed to the more theoretical meta level analysis for cut elimination.

6.1 IntHistGC

6.1.1 Global Caching

The main novelty of our provers.

6.2 IntLSJ

Chapter 7

Conclusion

7.1 Why this is a Very Clever Thesis

Appendix A

Appendix

A.1 Code